~~Patent~~ We claims

1. A method for the computer-aided interchange of cryptographic keys between a first computer unit (U) and a second computer unit (N),

- in which a first value ($g^t$) is formed from a first random number (t) using a generating element (g) of a finite group in the first computer unit (U),
- in which a first message (M1) is transmitted from the first computer unit (U) to the second computer unit (N), the first message (M1) containing at least the first value ($g^t$),
- in which a session key (K) is formed in the second computer unit (N) using a first hash function (h1), a first input variable for the first hash function (h1) containing at least one first term which is formed by exponentiation of the first value ($g^t$) using a secret network key (s),
- in which the session key (K) is formed in the first computer unit (U) using the first hash function (h1), a second input variable for the first hash function (h1) containing at least one second term which is formed by exponentiation of a public network key ($g^s$) using the first random number (t),
- in which a fourth input variable is formed in the first computer unit (U) using a second hash function (h2) or the first hash function (h1), a third input variable for the first hash function (h1) or for the second hash function (h2) containing, for the purpose of forming the fourth input variable, one or more variables which can be used to infer the session key unambiguously,

- in which a signature term is formed in the first computer unit (U) from at least the fourth input variable using a first signature function ($Sig_U$),
- in which a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing at least the signature term from the first computer unit (U), and
- in which the signature term is verified in the second computer unit (N).

2. The method as claimed in claim 1, in which the secret network key and/or the public network key is/are long-service keys.

3. The method as claimed in claim 1 or 2, in which the third input variable contains a plurality of variables which can be used to infer the session key unambiguously.

4. The method as claimed in one of claims 1 to 3, in which the variable or the variables contains or contain at least the first value ($g^t$) and/or the public network key ($g^s$).

5. The method as claimed in one of claims 1 to 4,
- in which the first message (M1) contains an identity statement ($id_{CA}$) for a certification computer unit (CA) which delivers a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
- in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the network certificate

(CertN) or the chain of certificates, the last of which is the network certificate (CertN), and

- in which the network certificate (CertN) or the chain of certificates, the last of which is the network certificate (CertN), is verified in the first computer unit (U).

6. The method as claimed in claim 5,

- in which a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing a user certificate (CertU) or a chain of certificates, the last of which is the user certificate (CertU),
- in which the user certificate (CertU) or the chain of certificates, the last of which is the user certificate (CertU), is verified in the second computer unit (N).

7. The method as claimed in one of claims 1 to 6,

- in which the first message(M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement ($id_{CA}$) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) which can be verified by the first computer unit (U),
- in which a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the first value ($g^t$) as input variable,
- in which a fifth message (M5), containing at least the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate

(CertU), is transmitted from the certification computer unit (CA) to the second computer unit (N).

8. The method as claimed in one of claims 1 to 7,

- in which a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the public network key ($g^s$), the first value ($g^t$), the identity variable (IMUI) for the first computer unit (U) as input variable, and an output variable from a third hash function (h3) being signed using a second signature function ($Sig_N$),
- in which the first signed term is verified in the certification computer unit (CA),
- in which a third term, containing at least the first value ($g^t$), the public network key ($g^s$) and an identity statement ($id_N$) for the second computer unit (N), is formed in the certification computer unit (CA),
- in which a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- in which the hash value for the third term is signed in the certification computer unit (CA) using a third signature function ($Sig_{CA}$),
- in which a network certificate (CertN) containing at least the third term and the signed hash value for the third term is formed in the certification computer unit (CA),
- in which a fourth hash function (h4) is applied in the certification computer unit (CA) to a fifth term, containing at least the identity statement ($id_N$) for the second computer unit (N) and a user certificate (CertU),
- in which the hash value for the fifth term is signed using the secret certification key (cs)

by using the third signature function ($Sig_{CA}$), and the result represents the second signed term,

- in which a fifth message (M5), containing at least the network certificate (CertN), the fifth term and the second signed term, is transmitted from the certification computer unit (CA) to the second computer unit (N),
- in which the network certificate (CertN) and the second signed term are verified in the second computer unit (N),
- in which a fourth term, containing at least the public network key ($g^s$) and the signed hash value for the third term, is formed in the second computer unit (N),
- in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the fourth term, and
- in which the network certificate (CertN) is verified in the first computer unit (U).

9. The method as claimed in one of claims 1 to 8,

- in which the first message (M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement ($id_{CA}$) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
- in which a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least one certificate for the public network key ($g^s$), the first value

($g^t$) and the identity variable (IMUI) for the first computer unit (U),

- in which a third term, containing at least the public network key ($g^s$) or a variable which determines the public network key ($g^s$) unambiguously, is formed in the certification computer unit (CA),
- in which a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- in which the hash value for the third term is signed in the certification computer unit (CA) using a third signature function ($Sig_{CA}$),
- in which a fifth message (M5), containing at least the signed hash value for the third term, is transmitted from the certification computer unit (CA) to the second computer unit (N),
- in which the signed hash value for the third term is verified in the second computer unit (N),
- in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the signed hash value for the third term, and
- in which the signed hash value for the third term is verified in the first computer unit (U).

10. The method as claimed in claim 9,
in which the third term contains a public user signature key (KU) or a variable which determines the user signature key (KU) unambiguously.

11. The method as claimed in claim 9 or 10,
in which the fifth message (M5) and the second message (M2) have at least one chain of certificates.

12. The method as claimed in claim 8, in which the fifth term has a time stamp (TS).

13. The method as claimed in one of claims 9 to 12, in which the third term has a time stamp (TS).

14. The method as claimed in one of claims 7 to 13,
- in which an intermediate key (L) is formed in the first computer unit (U), before formation of the first message (M1), by raising a public key declaration key ($g^u$) to a higher power using the first random number (t),
- in which a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the first message (M1), from the identity variable (IMUI) for the first computer unit (U) by encrypting the identity variable (IMUI) with the intermediate key (L) using an encryption function (Enc),
- in which the first message (M1) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U),
- in which the fourth message (M4) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U).

15. The method as claimed in one of claims 7 to 14, in which the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate (CertU), is encrypted with L in the fifth message (M5).

16. The method as claimed in one of claims 7 to 15,

in which at least one of the variables, the identity statement ($id_N$) for the second computer unit (N), the identity variable (IMUI) for the first computer unit (U), the public network key ($g^s$), the network certificate (CertN) or the user certificate (CertU) is checked in the certification computer unit (CA) using a revocation list.

17. The method as claimed in one of claims 1 to 16,
- in which the first message (M1) contains at least one old temporary identity variable (TMUIO) for the first computer unit (U),
- in which a new temporary identity variable (TMUIN) is formed for the first computer unit (U) in the second computer unit (N) after the first message (M1) has been received and before the second message (M2) is formed,
- in which a fifth encrypted term (VT5) is formed from the new temporary identity variable (TMUIN) for the first computer unit (U) by encrypting the new temporary identity variable (TMUIN) for the first computer unit (U) with the session key (K) using the encryption function (Enc),
- in which the second message (M2) contains at least the fifth encrypted term (VT5),
- in which the fifth encrypted term (VT5) is decrypted in the first computer unit (U) after the second message (M2) has been received and before the fourth input variable is formed,
- in which the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the new temporary identity variable (TMUIN) for the first computer unit (U) for the purpose of forming the fourth input variable, and

- in which the third message (M3) does not contain the identity variable (IMUI) for the first computer unit (U).

18. The method as claimed in one of claims 1 to 17,
- in which a response (A) containing information about the session key (K) is formed in the second computer unit (N),
- in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the response (A), and
- in which the session key (K) is checked in the first computer unit (U) using the response (A).

19. The method as claimed in one of claims 1 to 18, in which the third message (M3) contains an identity variable (IMUI) for the first computer unit (U).

20. The method as claimed in one of claims 1 to 19,
- in which the first input variable for the first hash function (h1) contains at least one second random number (r) in the second computer unit (N),
- in which the second message (M2) contains the second random number (r), and
- in which the second input variable for the first hash function (h1) contains at least the second random number (r) in the first computer unit (U).

21. The method as claimed in one of claims 1 to 20, in which the variable or the variables as claimed in claim 3 contains or contain the second random number (r).

22. The method as claimed in one of claims 1 to 21,

- in which a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the third message (M3), from the identity variable (IMUI) for the first computer unit (U) by encrypting at least the identity variable (IMUI) with the session key (K) using the encryption function (Enc),
- in which the third message (M3) contains the second encrypted term (VT2), and
- in which the second encrypted term (VT2) is decrypted in the second computer unit (N) after the third message (M3) has been received.

23. The method as claimed in one of claims 1 to 22,
- in which the second message (M2) contains an optional first data field (dat1), and
- in which the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the optional first data field (dat1) for the purpose of forming the fourth input variable.

24. The method as claimed in one of claims 1 to 23,
- in which a third encrypted term (VT3) is formed in the first computer unit (U), before formation of the third message (M3), by encrypting at least one optional second data field (dat2) with the session key (K) using the encryption function (Enc),
- in which the third message (M3) contains at least the third encrypted term (VT3), and
- in which the third encrypted term (VT3) is decrypted in the second computer unit (N) after the third message (M3) has been received.

25. The method as claimed in one of claims 1 to 24,
- in which a first encrypted term (VT1) is formed in the first computer unit (U), before formation

of the third message (M3), by encrypting at least the signature term using the encryption function (Enc),

- in which the third message (M3) contains the first encrypted term (VT1), and
- in which the first encrypted term (VT1) is decrypted in the second computer unit (N) after the third message (M3) has been received and before the signal term is verified.

26. The method as claimed in one of claims 1 to 25, in which a response (A) is formed in the second computer unit (N) by encrypting a constant (const), and possibly further variables, which are known in the second computer unit (N) and in the first computer unit (U), with the session key (K) using the encryption function (Enc).

27. The method as claimed in one of claims 1 to 26, in which the response (A) is checked in the first computer unit (U) by encrypting a constant (const), and possibly further variables, with the session key (K) using the encryption function (Enc) and comparing the result with the response (A).

28. The method as claimed in one of claims 1 to 26, in which the response (A) is checked in the first computer unit (U) by decrypting the response (A) with the session key (K) using the encryption function (Enc) and comparing a decrypted constant (const') with a constant (const), and possibly further variables.

29. The method as claimed in one of claims 1 to 28,

- in which a response (A) is formed in the second computer unit (N) by applying a third hash

function (h3) to an input variable which contains at least the session key (K), and

- in which the response (A) is checked in the first computer unit (U) by applying the third hash function (h3) to the input variable, which contains at least the session key (K), and comparing the result with the response (A).

30. The method as claimed in one of claims 1 to 29, in which the third message (M3) contains at least one optional second data field (dat2).

31. The method as claimed in one of claims 1 to 30, in which the first computer unit (U) is formed by a mobile communication terminal and/or the second computer unit (N) is formed by an authentication unit in a mobile communication network.

32. An arrangement for the computer-aided interchange of cryptographic keys between a first computer unit (U) and a second computer unit (N), in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- a first value ($g^t$) is formed from a first random number (t) using a generating element (g) of a finite group in the first computer unit (U),
- a first message (M1) is transmitted from the first computer unit (U) to the second computer unit (N), the first message (M1) containing at least the first value ($g^t$),
- a session key (K) is formed in the second computer unit (N) using a first hash function (h1), a first input variable for the first hash function (h1) containing at least one first term which is formed by exponentiation of the first value ($g^t$) using a secret network key (s),

- the session key (K) is formed in the first computer unit (U) using the first hash function (h1), a second input variable for the first hash function (h1) containing at least one second term which is formed by exponentiation of a public network key ($g^s$) using the first random number (t),
- a fourth input variable is formed in the first computer unit (U) using a second hash function (h2) or the first hash function (h1), a third input variable for the first hash function (h1) or for the second hash function (h2) containing, for the purpose of forming the fourth input variable, one or more variables which can be used to infer the session key unambiguously,
- a signature term is formed in the first computer unit (U) from at least the fourth input variable using a first signature function ($Sig_U$),
- a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing at least the signature term from the first computer unit (U), and
- the signature term is verified in the second computer unit (N).

33. The arrangement as claimed in claim 31, in which the secret network key and/or the public network key is/are long-service keys.

34. The arrangement as claimed in claim 32 or 33, in which the first computer unit (U) and the second computer unit (N) are set up such that the third input variable contains a plurality of variables which can be used to infer the session key unambiguously.

35. The arrangement as claimed in one of claims 32 to 34,
in which the first computer unit (U) and the second computer unit (N) are set up such that the variable or the variables contains or contain at least the first value ($g^t$) and/or the public network key ($g^s$).

36. The arrangement as claimed in one of claims 32 to 35,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the first message (M1) contains an identity statement ($id_{CA}$) for a certification computer unit (CA) which delivers a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the network certificate (CertN) or the chain of certificates, the last of which is the network certificate (CertN), and
- the network certificate (CertN) or the chain of certificates, the last of which is the network certificate (CertN), is verified in the first computer unit (U).

37. The arrangement as claimed in claim 36,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing a user certificate (CertU) or a chain of

certificates, the last of which is the user certificate (CertU),

- the user certificate (CertU) or the chain of certificates, the last of which is the user certificate (CertU), is verified in the second computer unit (N).

38. The arrangement as claimed in one of claims 32 to 37,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- the first message(M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement ($id_{CA}$) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) which can be verified by the first computer unit (U),
- a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the first value ($g^t$) as input variable,
- a fifth message (M5), containing at least the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate (CertU), is transmitted from the certification computer unit (CA) to the second computer unit (N).

39. The arrangement as claimed in one of claims 32 to 38,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the public network key ($g^s$), the first value ($g^t$), the identity variable (IMUI) for the first computer unit (U) as input variable, and an output variable from a third hash function (h3) being signed using a second signature function ($Sig_N$),
- the first signed term is verified in the certification computer unit (CA),
- a third term, containing at least the first value ($g^t$), the public network key ($g^s$) and an identity statement ($id_N$) for the second computer unit (N), is formed in the certification computer unit (CA),
- a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- the hash value for the third term is signed in the certification computer unit (CA) using a third signature function ($Sig_{CA}$),
- a network certificate (CertN) containing at least the third term and the signed hash value for the third term is formed in the certification computer unit (CA),
- a fourth hash function (h4) is applied in the certification computer unit (CA) to a fifth term, containing at least the identity statement ($id_N$) for the second computer unit (N) and a user certificate (CertU),
- the hash value for the fifth term is signed using the secret certification key (cs) by using the third signature function ($Sig_{CA}$), and the result represents the second signed term,
- a fifth message (M5), containing at least the network certificate (CertN), the fifth term and the second signed term, is transmitted from the

certification computer unit (CA) to the second computer unit (N),

- the network certificate (CertN) and the second signed term are verified in the second computer unit (N),
- a fourth term, containing at least the public network key ($g^s$) and the signed hash value for the third term, is formed in the second computer unit (N),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the fourth term, and
- the network certificate (CertN) is verified in the first computer unit (U).

40. The arrangement as claimed in one of claims 33 to 39,

in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- the first message (M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement ($id_{CA}$) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
- a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least one certificate for the public network key ($g^s$), the first value ($g^t$) and the identity variable (IMUI) for the first computer unit (U),

- a third term, containing at least one public network key ($g^s$) or a variable which determines the public network key ($g^s$) unambiguously, is formed in the certification computer unit (CA),
- a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- the hash value for the third term is signed in the certification computer unit (CA) using a third signature function ($Sig_{CA}$),
- a fifth message (M5), containing at least the signed hash value for the third term, is transmitted from the certification computer unit (CA) to the second computer unit (N),
- the signed hash value for the third term is verified in the second computer unit (N),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the signed hash value for the third term, and
- the signed hash value for the third term is verified in the first computer unit (U).

41. The arrangement as claimed in claim 40, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the third term contains the public user signature key (KU) or a variable which determines the user signature key (KU) unambiguously.

42. The arrangement as claimed in claim 40 or 41, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the fifth message (M5) and the second message (M2) contain at least one chain of certificates.

43. The arrangement as claimed in one of claims 38 to 42,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the fifth term has a time stamp (TS).

44. The arrangement as claimed in one of claims 38 to 43,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the third term has a time stamp (TS).

45. The arrangement as claimed in claims 38 to 44,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- an intermediate key (L) is formed in the first computer unit (U), before formation of the first message (M1), by raising a public key declaration key ($g^u$) to a higher power using the first random number (t),
- a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the first message (M1), from the identity variable (IMUI) for the first computer unit (U) by encrypting the identity variable (IMUI) with the intermediate key (L) using an encryption function (Enc),
- the first message (M1) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U),
- the fourth message (M4) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U).

46. The arrangement as claimed in one of claims 38 to 45,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate (CertU), is encrypted with L in the fifth message (M5).

47. The arrangement as claimed in one of claims 38 to 46,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
at least one of the variables, the identity statement ($id_N$) for the second computer unit (N), the identity variable (IMUI) for the first computer unit (U), the public network key ($g^s$), the network certificate (CertN) or the user certificate (CertU) is checked in the certification computer unit (CA) using a revocation list.

48. The arrangement as claimed in one of claims 32 to 47,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the first message (M1) contains at least one old temporary identity variable (TMUIO) for the first computer unit (U),
- a new temporary identity variable (TMUIN) is formed for the first computer unit (U) in the second computer unit (N) after the first message

(M1) has been received and before the second message (M2) is formed,

- a fifth encrypted term (VT5) is formed from the new temporary identity variable (TMUIN) for the first computer unit (U) by encrypting the new temporary identity variable (TMUIN) for the first computer unit (U) with the session key (K) using the encryption function (Enc),
- the second message (M2) contains at least the fifth encrypted term (VT5),
- the fifth encrypted term (VT5) is decrypted in the first computer unit (U) after the second message (M2) has been received and before the fourth input variable is formed,
- the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the new temporary identity variable (TMUIN) for the first computer unit (U) for the purpose of forming the fourth input variable, and
- the third message (M3) does not contain the identity variable (IMUI) for the first computer unit (U).

49. The arrangement as claimed in one of claims 32 to 48,

in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- a response (A) containing information about the session key (K) is formed in the second computer unit (N),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the response (A), and
- the session key (K) is checked in the first computer unit (U) using the response (A).

50. The arrangement as claimed in one of claims 32 to 49,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the third message (M3) contains an identity variable (IMUI) for the first computer unit (U).

51. The arrangement as claimed in one of claims 32 to 48,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the first input variable for the first hash function (h1) contains at least one second random number (r) in the second computer unit (N),
- the second message (M2) contains the second random number (r), and
- the second input variable for the first hash function (h1) contains at least the second random number (r) in the first computer unit (U).

52. The arrangement as claimed in one of claims 32 to 47,
in which the first computer unit (U) and the second computer unit (N) are set up such that the variable or the variables as claimed in claim 34 contains or contain the second random number (r).

53. The arrangement as claimed in one of claims 32 to 51,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the

third message (M3), from the identity variable (IMUI) for the first computer unit (U) by encrypting at least the identity variable (IMUI) with the session key (K) using the encryption function (Enc),

- the third message (M3) contains the second encrypted term (VT2), and
- the second encrypted term (VT2) is decrypted in the second computer unit (N) after the third message (M3) has been received.

54. The arrangement as claimed in one of claims 32 to 53,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- the second message (M2) contains an optional first data field (dat1), and
- the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the optional first data field (dat1) for the purpose of forming the fourth input variable.

55. The arrangement as claimed in one of claims 32 to 54,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- a third encrypted term (VT3) is formed in the first computer unit (U), before formation of the third message (M3), by encrypting at least one optional second data field (dat2) with the session key (K) using the encryption function (Enc),
- the third message (M3) contains at least the third encrypted term (VT3), and

- the third encrypted term (VT3) is decrypted in the second computer unit (N) after the third message (M3) has been received.

56. The arrangement as claimed in one of claims 32 to 55,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a first encrypted term (VT1) is formed in the first computer unit (U), before formation of the third message (M3), by encrypting at least the signature term using the encryption function (Enc),
- the third message (M3) contains the first encrypted term (VT1), and
- the first encrypted term (VT1) is decrypted in the second computer unit (N) after the third message (M3) has been received and before the signal term is verified.

57. The arrangement as claimed in one of claims 32 to 56,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
a response (A) is formed in the second computer unit (N) by encrypting a constant (const), and possibly further variables, which are known in the second computer unit (N) and in the first computer unit (U), with the session key (K) using the encryption function (Enc).

58. The arrangement as claimed in one of claims 44 to 57,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

the response (A) is checked in the first computer unit (U) by encrypting a constant (const), and possibly further variables, with the session key (K) using the encryption function (Enc) and comparing the result with the response (A).

59. The arrangement as claimed in one of claims 44 to 57,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the response (A) is checked in the first computer unit (U) by decrypting the response (A) with the session key (K) using the encryption function (Enc) and comparing a decrypted constant (const'), and possibly further variables, with a constant (const).

60. The arrangement as claimed in one of claims 32 to 59,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a response (A) is formed in the second computer unit (N) by applying a third hash function (h3) to an input variable which contains at least the session key (K), and
- the response (A) is checked in the first computer unit (U) by applying the third hash function (h3) to an input variable, which contains at least the session key (K), and comparing the result with the response (A).

61. The arrangement as claimed in one of claims 32 to 60,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

the third message (M3) contains at least one optional second data field (dat2).

62. The arrangement as claimed in one of claims 32 to 61,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the first computer unit (U) is formed by a mobile communication terminal and/or the second computer unit (N) is formed by an authentication unit in a mobile communication network.